

Communication Links for Distributed Quantum Computation

Rodney Van Meter, *Member, IEEE*, Kae Nemoto, and W.J. Munro

Abstract—Distributed quantum computation requires quantum operations that act over a distance on error-correction encoded states of logical qubits, such as the transfer of qubits via teleportation. We evaluate the performance of several quantum error correction codes, and find that teleportation failure rates of one percent or more are tolerable when two levels of the $[[23,1,7]]$ code are used. We present an analysis of performing quantum error correction (QEC) on QEC-encoded states that span two quantum computers, including the creation of distributed logical zeroes. The transfer of the individual qubits of a logical state may be multiplexed in time or space, moving serially across a single link, or in parallel across multiple links. We show that the performance and reliability penalty for using serial links is small for a broad range of physical parameters, making serial links preferable for a large, distributed quantum multicomputer when engineering difficulties are considered. Such a multicomputer will be able to factor a 1,024-bit number using Shor's algorithm with a high probability of success.

I. INTRODUCTION

DISTRIBUTED quantum computation uses the physical resources of two or more quantum computers to solve a single problem [1]–[6]. These computers may be geographically distributed, or may be colocated, with the distributed nature of the system used to overcome the inherent limitations on the size of a single quantum computer [7], [8]. Distributed quantum computation naturally depends on the development of quantum networking technology to connect the computers [9], [10].

A *quantum computer* is a device that uses non-classical, quantum behavior of some physical phenomena to calculate certain functions asymptotically faster than a purely classical machine can [11], [12]. The fundamental unit of data in a quantum computer is a *qubit*, which has two possible states, written $|0\rangle$ and $|1\rangle$, analogous to the 0 and 1 of a classical bit. These states may be the horizontal and vertical polarization of a photon, the up and down spin of a single electron, or the direction of a single quantum of magnetic flux; dozens of quantum phenomena have been proposed as qubits, and many of them are under experimental evaluation [7], [11], [12]. Most systems, with the obvious exception of photons, hold qubits in a register, and execute “gates” on the qubits, manipulating their state like instructions in a classical computer manipulate the bits of a register.

Perhaps the three most famous quantum algorithms are Shor's algorithm for factoring large numbers, Grover's search

algorithm, and the Deutsch-Jozsa algorithm for distinguishing among certain classes of functions [13]–[15]. Shor's algorithm appears to offer a superpolynomial speedup for factoring, compared to the best known classical algorithm. Grover has shown that, for unstructured search problems, the best a quantum computer can do is to search all N possible solutions in $O(\sqrt{N})$ operations, while Deutsch-Jozsa turns a probabilistic problem into one with a deterministic, certain answer after a single iteration. All three have been demonstrated experimentally at very small scales [16]–[18]. However, designing and building quantum computers capable of solving problems at scales that are classically intractable will require many more years of effort from physicists working on the basic technologies, theorists designing algorithms including quantum error correction, and quantum computer architects.

Quantum computation utilizes the quantum characteristics of *superposition*, *entanglement*, *quantum interference*, and *measurement* to achieve its speedup in computational class. Superposition, entanglement and interference refer to the wavelike behavior of a quantum system. For our qubit, we have two basis states, $|0\rangle$ and $|1\rangle$, which can be distinguished by measurement in the computational basis, giving a classical value. A superposition state contains amplitudes for $|0\rangle$ and $|1\rangle$ *at the same time*. For instance, the superposition $|0\rangle + |1\rangle$ has equal amplitudes for each basis state, meaning that there is a 50% probability of measuring the qubit in $|0\rangle$ and a 50% probability in $|1\rangle$. Superpositions of quantum states are the source of the interference that drives a quantum computer; quantum algorithms attempt to manipulate the *amplitude and phase* of various states so that desirable states (the answers to the problem being solved) have a high probability of being measured while the undesirable states (the non-answers to the problem being solved) have a low probability of being measured.

Superposition can extend beyond single qubits and can be seen in multi-qubit situations. Two qubits (labelled A and B) can exist in a quantum state such as

$$|\psi\rangle_{AB} = |0\rangle_A |1\rangle_B - |1\rangle_A |0\rangle_B. \quad (1)$$

In this interesting state, if we measure the first qubit to be in the state $|0\rangle_A$, then the second qubit has to be in the state $|1\rangle_B$; conversely, getting the measurement result $|1\rangle_A$ guarantees that we will find $|0\rangle_B$. The A and B measured results are perfectly anti-correlated. This multi-qubit superposition described above is generally given the special name *entanglement* because neither qubit can be said to be in a state of its own, independent of the other. The state cannot be

R. Van Meter is with National Institute of Informatics, Tokyo, Japan, and with CREST-JST. K. Nemoto is with National Institute of Informatics, Tokyo, Japan. W.J. Munro is with HP Labs, Bristol, UK.

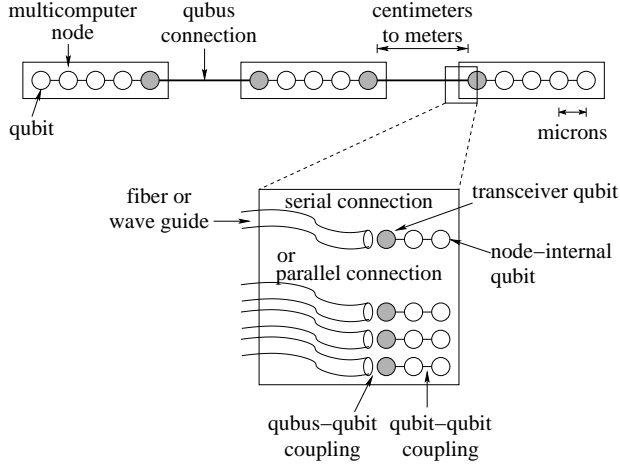


Fig. 1. A quantum multicomputer architecture with detail of qubus connections.

factored into a product, $|\psi\rangle_{AB} \neq |\chi\rangle_A |\eta\rangle_B$, for any choice of basis transformation. Many gates that act on two qubits can change their level of entanglement, increasing or decreasing it, depending on the gate and the initial state of the qubits. Once a pair of qubits are entangled, they may be separated by any distance, and will retain their shared state. This behavior results in the “spooky action at a distance” that so disturbed Einstein about quantum theory. The maximally entangled pair of qubits are called *EPR pairs* or *Bell pairs*, and can be used to *teleport* quantum data, such that the unknown state of one qubit can be moved from one location to another without transporting the physical carrier of information of the qubit, consuming an EPR pair in the process [19], [20].

It has been shown that entanglement between the separate quantum computers, or nodes, of a distributed quantum system is necessary if the system is to have the potential for exponential speedup over a classical computer (or cluster of classical computers) [21]–[23]. At a practical level, this need for node-spanning entanglement arises because application algorithms require gates that act on data that is stored in separate nodes [8], [24], [25]. This can be achieved by teleporting data from node to node and performing computation locally (which we refer to as *teledata*), or, alternatively, by using essentially the same techniques to execute the equivalent of a local gate over a distance, without bringing the two qubits together. This technique is known as teleporting a gate (which we refer to as *telegate*) [26]. We have found that, for some application workloads and a reasonable set of physical assumptions, it is better to teleport data than gates [8], [24].

Our *quantum multicomputer* (QMC) architecture is composed of many small nodes, holding only a few logical qubits each, with each node connected to two neighbors, left and right, into a line, as shown in Figure 1. The connections are assumed to be *qubus* links, which entangle distant qubits using a strong probe laser beam that interacts weakly with qubits connected to the bus, which we call the *transceiver qubits* [27]–[29]. The connections could also be made using single photons instead of the qubus [3], [30]–[45], though the

TABLE I
NUMBER OF TELEPORTATIONS NECESSARY TO EXECUTE THE FULL MODULAR EXPONENTIATION FOR SHOR’S FACTORING ALGORITHM FOR DIFFERENT PROBLEM SIZES ON A QUANTUM MULTICOMPUTER [8], [24].

length	teleportations (t)
16	14000–125000
128	8×10^6 – 10^8
1024	4×10^9 – 6×10^{10}

basic architecture is independent of this choice. The qubits may be solid-state qubits (such as quantum dots or one of several types of Josephson junction superconducting qubits), and are assumed to be capable of interacting with their neighbors inside a single node or with the qubus.

The performance of any computing system must be measured with respect to a particular workload; we have found that this configuration works well for Shor’s factoring algorithm [13]. The most computationally intensive portion of the algorithm is the modular exponentiation [46]–[48]. This modular exponentiation is $O(n^3)$ for factoring an n -bit number, both in local gate count and in teleportation operations. Table I shows the number of logical qubit teleportations necessary to execute the modular exponentiation portion of Shor’s algorithm for 16, 128, and 1,024 bits. The design choices of the number of qubits per node and the addition algorithm to be used are important. The carry-lookahead adder requires ten to fifteen times as many teleportations as the carry-ripple adders (for 16 to 1,024 bits), but may produce results faster under some circumstances; this accounts for the range of values in Table I [48]–[50]. The numbers in this table are used to choose the values for the results presented in Table III.

Individual physical qubits are quite fragile and prone to errors and deterioration over time; therefore, application-level algorithms are generally assumed to run on logical qubits, encoded in multiple physical qubits via *quantum error correction* [51]–[55]. Such error codes are generally described as $[[n, k, d]]$ codes, where n is the number of lower-level qubits in a block, k is the number of logical qubits the block represents, and $(d - 1)/2$ is the maximum number of errors in the block that will not corrupt the state. The coding efficiency k/n of quantum codes is lower than classical codes because quantum states must be protected from errors in both value and phase, as well as being inherently more delicate than classical states. Research has concentrated on $k = 1$ codes both because simulating larger systems is difficult, and because executing logical operations on $k = 1$ encoded states is substantially easier than $k > 1$ states. Codes discovered early in the development of quantum computing include the $[[7, 1, 3]]$ code based on a Hamming code, the quantum-unique $[[5, 1, 3]]$ code, and the $[[9, 1, 3]]$ code derived from the simplest classical triple-redundancy protocol. More recently, Steane has been investigating larger known classes of classical codes for their quantum suitability and has recommended a $[[23, 1, 7]]$ code based on a Golay code [56], and Brun *et al.* have shown how to ease some of the restrictions on the choice of code by utilizing entanglement [57]. In this paper, we examine the interaction

of the $[[7,1,3]]$ and $[[23,1,7]]$ codes with the teleportation necessary for distributed quantum computation. Because the encoded states of the $[[7,1,3]]$ code are easier to manipulate than the states of the $[[5,1,3]]$ code, it is generally considered more attractive. The $[[23,1,7]]$ code is efficient relative to the strength of protection provided, as we will show in Section IV. Therefore, we focus on these two codes.

This paper addresses two issues relevant to the design of systems for distributed quantum computation: the necessary strength of error correction to provide a high probability of success of a lengthy but finite computation when teleportation is used as described above; and whether the quantum error correction-encoded block may be transmitted serially or must be transmitted in parallel, which helps determine our hardware design. Section II describes how distributed logical zero states can be constructed, providing the basis for doing error correction on logical states that span multiple nodes. Section III shows the use of distributed logical zeroes in maintaining distributed states and performing the error correction while the states are in motion. The next section discusses how different error correction codes improve the allowable teleportation error rate, assuming that each logical qubit is teleported in its entirety as necessary. Section V shows that serial links perform nearly as well as parallel links, before we conclude in Section VI.

II. DISTRIBUTED LOGICAL ZEROES

Figure 2 shows a circuit for taking seven qubits initialized to zero and combining them into a logical zero state ($|0\rangle_L$) for the Steane $[[7,1,3]]$ quantum error correcting code. This state is used in the fault-tolerant construction of quantum error correction and in fault-tolerant logical gates on encoded states. In distributed quantum computation, we may need to perform QEC on states that span two (or more) nodes, such as during data movement between nodes in a quantum multicomputer, or to maintain the integrity of a static state that spans multiple nodes. Thus, we must find a way to either

- 1) create a distributed $|0\rangle_L$ state;
- 2) do parity (error syndrome) measurements using only the qubits's weak nonlinearity approach or single photons on four or more qubits; or
- 3) find some other way to do syndrome measurements without the full, distributed $|0\rangle_L$ state.

Of these three options, we have chosen the first. We have also invested some effort in looking for a way to calculate the parity of n qubits using the weak nonlinearity, but all of the schemes we have found so far for more than three qubits scale poorly in terms of noise; Yamaguchi *et al.* have designed a method that works for three qubits but not more [58]. Bacon has developed a new method for creating self-correcting memories, using the original Shor $[[9,1,3]]$ code, that may not require the creation of logical zeroes; its implications for actual implementation are exciting but still poorly understood [59], [60]. Thus, $|0\rangle_L$ states must be created, and this section discusses the performance and error characteristics of the creation process.

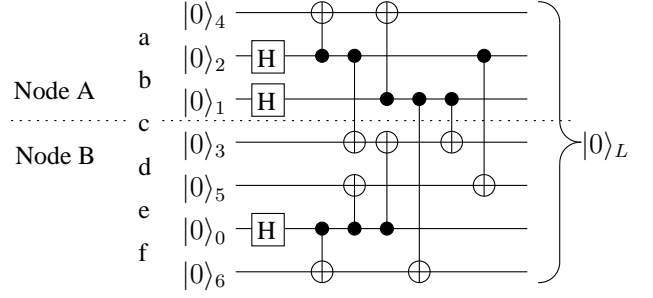


Fig. 2. Distributed circuit to create the $|0\rangle_L$ state for the Steane $[[7,1,3]]$ code.

TABLE II
BREAKPOINTS (CORRESPONDING TO FIGURE 2) AND THE COST OF TELEGATE V. TELEDATA TO CREATE A LOGICAL ZERO STATE FOR THE STEANE $[[7,1,3]]$ CODE, IN EPR PAIRS CONSUMED. THE DIRECTION QUBITS MUST BE TELEPORTED IS ALSO SHOWN FOR TELEDATA.

breakpoint	telegate	teledata
a	2	1 ($B \rightarrow A$)
b	3	2 ($B \rightarrow A$)
c	4	3 ($B \rightarrow A$)
d	3	3 ($A \rightarrow B$)
e	3	2 ($A \rightarrow B$)
f	2	1 ($A \rightarrow B$)

The logical $|0\rangle_L$ can be created using the same two methods as any other distributed quantum computation: we can directly create the state in a distributed fashion, using teleported gates (telegate), or we can create the state within a single node and teleport several of the qubits to the remote node before using the state in our QEC (teledata). First, consider the use of teleported gates to create the $|0\rangle_L$ state. Figure 2 shows that splitting the $|0\rangle_L$ state across two nodes, as at the line labeled “c”, forces the execution of four teleported CNOTs, consuming four EPR pairs; breaking at “d” would require only three. In the figure, the subscripts represent the bit number in the QEC block; the qubits have been reordered compared to the common representation for efficiency. Our second alternative is to teleport portions of a locally-created $|0\rangle_L$ state. If enough qubits and computational resources are available at both nodes, we are free to create the state in either location and teleport some of the qubits; thus, the maximum number of qubits that must be teleported is $\lceil n/2 \rceil$, or 3 for the 7-bit Steane code. Table II shows the number of gate or data teleportations necessary, depending on the breakdown of qubits to nodes, showing that teledata requires the same or fewer EPR pairs, and so is preferred.

III. DISTRIBUTED DATA

A. Static Distributed States

If a logical data qubit $|\psi\rangle_L$ is split between nodes A and B in the same fashion as Figure 2, we will use distributed $|0\rangle_L$ states to calculate the syndromes for the error correction. Each syndrome calculation consumes one $|0\rangle_L$ state, first executing some gates to entangle it with the logical data qubit, then measuring the zero state. The $[[7,1,3]]$ code

requires six syndrome measurements (three “value” and three “phase” measurements), and Steane recommends measuring each syndrome at least twice, so each QEC cycle consumes at least a dozen logical zero states. With $|\psi\rangle_L$ divided at the “d” point, each $|0\rangle_L$ creation requires three teleportations, for a total of $3 \times 12 = 36$ EPR pairs destroyed to execute a single, full cycle of QEC.

The split described here allows a single logical qubit plus its QEC ancillae, a total of fourteen physical qubits, to be split between two nodes. The same principles apply to states split among a larger number of nodes, potentially allowing significantly smaller nodes to be useful, or allowing larger logical encoding blocks to be used, spread out among small, fixed-size nodes. More importantly for our immediate purposes, this analysis serves as a basis for considering the movement of logical states from node to node.

B. States in Motion

When considering the teleportation of logical qubits and their error correction needs, two general approaches are possible:

- 1) Transfer the entire QEC block, then perform QEC locally at the destination; or
- 2) use one of the methods described above for distributed QEC *between* the teleportations of the component qubits.

The analysis in Section IV assumes the first approach, which is conceptually simpler; does the second approach, shown in Figure 3, offer any advantages in either performance or failure probability? Using this approach, we attempt to reduce the overall error probability by incrementally correcting the logical state as it is teleported; to teleport the seven-bit state we perform local QEC before beginning, then do distributed QEC after each of the first six teleportations, then local QEC again after the seventh teleportation. Each distributed QEC (DQEC) block performs twelve distributed syndrome measurements. We can again choose telegate or teledata for the $|0\rangle_L$ state creation; the figure illustrates teledata. Using telegate, we would need the sum of the telegate column in Table II, or $2+3+4+3+3+2 = 17$, inter-node gates, for each syndrome that must be measured. To perform twelve measurements we consume a total of $12 \times 17 = 204$ EPR pairs. Using teledata, we would need only $1 + 2 + 3 + 3 + 2 + 1 = 12$ per syndrome, or 144 EPR pairs for the full twelve syndromes in a cycle. The worst-case DQEC block is $3 \times 12 = 36$ teleportations. Obviously, the probability of error is higher for 36 teleportations than for seven. Therefore, unless someone develops a means of measuring syndromes without using the $|0\rangle_L$ states, this second approach does not achieve its goal of reducing the total error probability. Performance-wise, the penalty for doing step-wise QEC is also stiff; we conclude that this approach is not useful, given our current knowledge.

IV. TELEPORTATION FAILURE RATES

Teleportation is composed of several phases: EPR pair creation, local gates, measurements, and classical communication. The EPR pairs necessary for teleportation can be created over

a fiber, interacting with a qubit at each end via single-photon methods, or a qubus that utilizes a strong probe beam and a weak nonlinearity, as noted above. Until we take up the issue of link design in Section V, we will assume that local gates, memory, and measurements are perfect, or at least much better than EPR pair creation. Therefore, when we talk about limits on the failure rate of teleportation, we are really referring to the fidelity (quality) of the EPR pair. The quality can be improved via purification [3], [30], [61], [62], which has a cost logarithmic in the starting fidelity; in this paper, we will not pursue further the best way to achieve EPR pairs of the necessary quality, though our results here may help to establish the target fidelity for qubit purification.

The argument here falls much along the lines of the threshold argument for quantum computation in general [63], [64]. Because we are dealing with a small number of levels of concatenation and a finite computation, we are less interested in the threshold itself than a specific calculation of the success probability for a chosen arrangement. A more detailed estimate considering all three separate error sources in memory, local gates, and teleportation, along the lines of Steane’s simulations [56] would differ slightly; here we restrict ourselves to a simple analysis involving teleportation errors only, while in later sections we will introduce memory errors, as well.

First, let us briefly consider the failure probability assuming no error correction on our qubits. The probability of success of the entire computation, then, rests on the success of *all* of the individual teleportation operations. If t is the total number of teleportations we must execute for the complete computation and p_t is the probability of failure of a single teleportation, our success probability is

$$p_s = (1 - p_t)^t = 1 - \binom{t}{1}p_t + \binom{t}{2}(-p_t)^2 \cdots \approx 1 - tp_t \quad (2)$$

for $tp_t \ll 1$. Our failure probability grows linearly with the number of teleportations we must execute, requiring $p_t \ll 1/t$. Error rates of 10^{-5} to 10^{-11} are unlikely to be experimentally achievable in the near future, so we quickly conclude that error correction on the logical states being transferred is necessary.

We have examined one-level QEC and two-level concatenated QEC. We have evaluated all of the one- and two-layer combinations of $[[7,1,3]]$ and $[[23,1,7]]$. For $p_t \ll 1$, most failures will occur in the lowest failure mode, $((d-1)/2) + 1 = (d+1)/2$ errors. We will approximate our total failure probability as the probability of $(d+1)/2$ errors occurring. The $[[7,1,3]]$ code can restore the correct state only when at most one component qubit has been corrupted. The $[[23,1,7]]$ code can defend against three errors, so we are interested in the probability of two and four errors, respectively, when using these codes.

Transferring the seven-qubit error correction code word from one quantum computer node to another, illustrated in Figure 4, consumes seven EPR pairs. The probability of m errors occurring is

$$p_e(n, m) = \binom{n}{m} (1 - p_t)^{n-m} p_t^m \approx \binom{n}{m} p_t^m \quad (3)$$

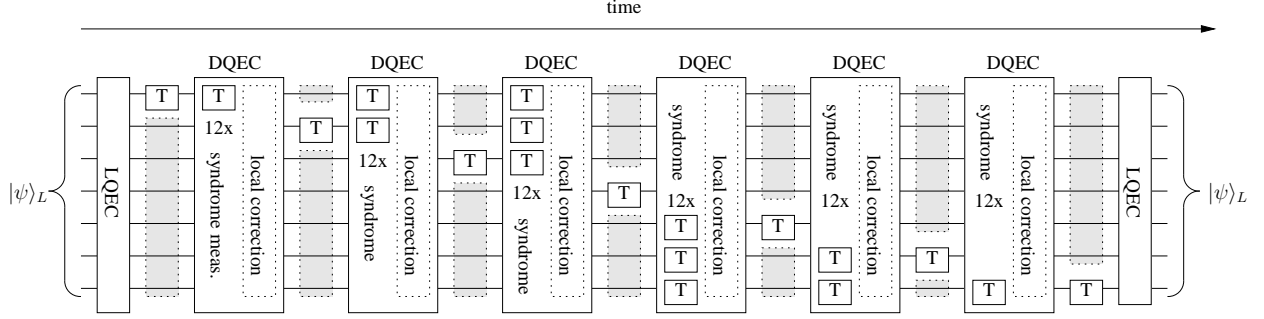


Fig. 3. Teleporting logical state using intermediate, teledata distributed QEC.

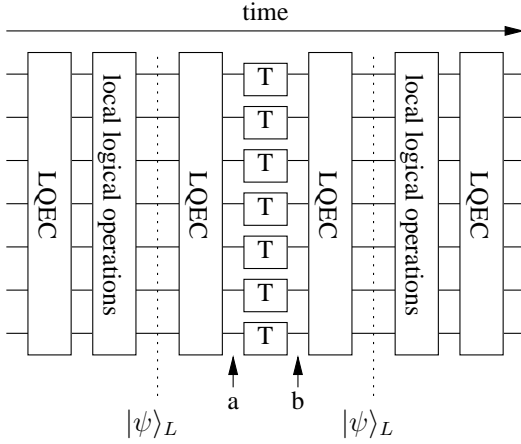


Fig. 4. Spatially multiplexed, block-level circuit for teleporting logical state using local QEC only, no intermediate QEC, over a parallel interface. The box holding a “T” is the teleportation circuit. Each line represents a qubit variable, independent of its location, so that the teleportation operation does not explicitly show the movement of the qubit from one node to another. LQEC, local quantum error correction.

for small p_t .

If p_f is the failure probability of our total algorithm and t is the *total* number of *logical* qubit teleportations we use in the computation, then

$$p_f = 1 - p_s = 1 - (1 - p_e)^t \approx \binom{t}{1} p_e = t p_e. \quad (4)$$

For this approximation to be valid, we require $t p_e \ll 1$. For the $[[7,1,3]]$ code,

$$p_e(7, 2) = \binom{7}{2} (1 - p_t)^5 p_t^2 \approx 21 p_t^2 \quad (5)$$

is the probability of two errors occurring in our block of seven qubits. Two qubit errors, of course, is more than the $[[7,1,3]]$ code can correct. Our probability of algorithm failure becomes

$$p_f \approx t p_e = 21 t p_t^2. \quad (6)$$

Thus, we can say that, to have a reasonable probability of success, we should have $p_t \ll 1/\sqrt{21t}$. This is a significant improvement over the case with no error correction seen above, but is still a stringent physical condition to meet if

t is large. For the stronger $[[23,1,7]]$ code,

$$p_e(23, 4) = \binom{23}{4} (1 - p_t)^{19} p_t^4 \approx 8855 p_t^4 \quad (7)$$

implying a desired $p_t \ll 1/\sqrt[4]{8855t} = 1/9.7 \sqrt[4]{t}$.

For two levels of the $[[7,1,3]]$ code, our total encoding will consist of seven blocks of seven qubits each, and the computation will fail only if *two* or more of those blocks fail. Of course, when using concatenation, the two codes need not be the same. Adapting Steane’s terminology and notation, will refer to the physical-level code as the “inner” code, and the code built on top of that as the “outer” code [56]. $[[n^i, k^i, d^i]]$ or $[[n, k, d]]^i$ is the inner code, and $[[n^o, k^o, d^o]]$ or $[[n, k, d]]^o$ is the outer code. Approximating the error probability according to Equations 3 and 4, we have

$$p_f \approx t \binom{n^o}{m^o} \left(\binom{n^i}{m^i} p_t^{m^i} \right)^{m^o} \quad (8)$$

where $m^i = (d^i + 1)/2$ and likewise for m^o .

Table III shows the estimates for the teleportation failure probability p_t that will give us a total algorithm failure probability of $p_f \leq 0.1$. The column titled “scale-up” is the number of physical qubits necessary to represent a logical qubit. Although $[[23,1,7]]^i + [[7,1,3]]^o$ and $[[7,1,3]]^i + [[23,1,7]]^o$ are different, by coincidence, their failure probabilities are almost identical. Note that $[[23,1,7]]$ offers essentially the same error protection as $[[7,1,3]]^i + [[7,1,3]]^o$, despite using half the number of qubits and being conceptually simpler.

From this analysis, we see that teleportation errors of 1% or more allow factoring of a 1,024-bit number on a quantum multicomputer. In this multicomputer, each of the 1,024 nodes contains nine logical qubits at a scale-up of 529, for a total of almost 5,000 physical qubits per node, when the Vedral-Barenco-Ekert (VBE) modular exponentiation algorithm is used. Seven of these logical qubits are used for the VBE algorithm, and one as a buffer for each teleportation link. Requirements for additional ancillae used for fault tolerance may increase the needed number of physical qubits by an amount dependent on the speed of the underlying technology at creating high-quality zero states and the need for local error correction.

TABLE III

AN ESTIMATE OF THE NECESSARY ERROR RATE OF TELEPORTATION (p_t) TO ACHIEVE A SPECIFIC NUMBER OF LOGICAL TELEPORTATIONS WITH 90% PROBABILITY OF SUCCESS ($p_f = 0.1$) FOR THE *entire* COMPUTATION, FOR DIFFERENT ERROR-CORRECTION SCHEMES.

error-correcting code	scale-up	teleportations (t)	allowable teleportation error rate p_t for $p_f = 0.1$
(none)	1	10^5 10^8 10^{11}	$p_t \leq p_f/t = 0.1/t = 10^{-6}$ 10^{-9} 10^{-12}
[[7,1,3]]	7	10^5 10^8 10^{11}	$p_t \leq \sqrt{p_f/21t} = \sqrt{0.1/21t} = 2.2 \times 10^{-4}$ 7×10^{-6} 2.2×10^{-7}
[[23,1,7]]	23	10^5 10^8 10^{11}	$p_t \leq \sqrt[4]{p_f/8855t} = 1/17 \sqrt[4]{t} \approx 3.3 \times 10^{-3}$ 5.8×10^{-4} 1×10^{-4}
[[7,1,3]] ⁱ +[[7,1,3]] ^o	49	10^5 10^8 10^{11}	$p_t \leq \sqrt{\sqrt{p_f/21t}/21} = 0.057/t^{1/4} \approx 3.2 \times 10^{-3}$ 5.7×10^{-4} 1×10^{-4}
[[23,1,7]] ⁱ +[[7,1,3]] ^o	161	10^5 10^8 10^{11}	$p_t \leq \sqrt[4]{\sqrt{p_f/21t}/8855} = 0.053/t^{1/8} \approx 0.013$ 5.3×10^{-3} 2.2×10^{-3}
[[7,1,3]] ⁱ +[[23,1,7]] ^o	161	10^5 10^8 10^{11}	$p_t \leq \sqrt[4]{\sqrt{p_f/8855t}/21} = 0.053/t^{1/8} \approx 0.013$ 5.3×10^{-3} 2.2×10^{-3}
[[23,1,7]] ⁱ +[[23,1,7]] ^o	529	10^5 10^8 10^{11}	$p_t \leq \sqrt[4]{\sqrt{p_f/8855t}/8855} = 0.051/t^{1/16} \approx 0.025$ 0.016 0.010

V. IMPLICATIONS FOR LINK DESIGN

The performance of error correction influences an important hardware design decision: should our network links be serial or parallel? We can multiplex the transfer of the qubits either temporally or spatially, as shown in Figure 1. The figure shows qubus fibers or wave guides coupling to one or more qubits. In the figure, the fiber and qubit are drawn approximately the same size, but in reality the fiber or wave guide is likely to be many times the size of the qubit. Thus, these connections may require large amounts of die space, force large qubit-qubit spacing (which affects the quality of interaction for some types of qubits), and make high-quality connections difficult, reducing manufacturing yield. Each qubus connection is therefore expensive, and minimizing their number is desirable. We argue that the difference in both reliability and performance is likely to be small, assuming that the reliability of teleportation is less than that of quantum memory and that teleportation times are reasonable compared to the cycle time of locally-executed QEC.

Figure 4 shows a [[7,1,3]] state being transferred in parallel and Figure 5 shows the serial equivalent. In these diagrams, each line represents a qubit that is a member of a code block, essentially following the variable rather than the storage locations; at a T block, representing teleportation, of course the qubit moves from one node to the other. If the transfer is done serially, the wait to *start* the QEC sequence is seven times as long, but the *total* time for transfer plus QEC (that is, time from the start of one QEC cycle to the next, from the first $|\psi\rangle_L$ to the point marked “b” in the figures) won’t grow by nearly as large a factor if local QEC requires significant time compared to a teleportation. Thus, we need to determine

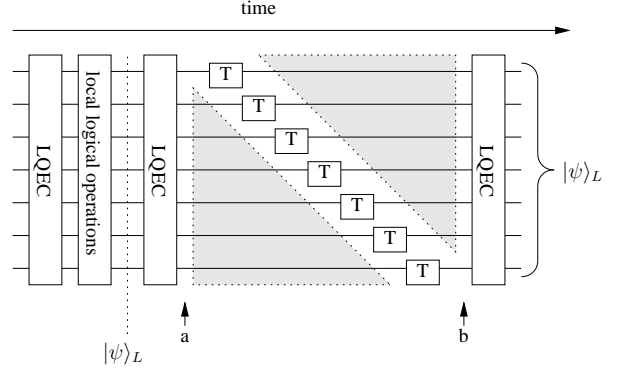


Fig. 5. Temporally multiplexed, logical circuit for encoded state transfer over a serial interface. LQEC, local quantum error correction. Teleporting logical state using local QEC only, no intermediate QEC. The box holding a “T” is the teleportation circuit. Each line represents a qubit variable, independent of its location, so that the teleportation operation does not explicitly show the movement of the qubit from one node to another.

if the increase in wait time caused by the lengthening of the interval from the point marked “a” to the point marked “b” in Figures 4 and 5 has an unacceptably large impact on our overall failure rate.

The gray areas in Figure 5 indicate increased wait time for the qubits. They total $n(n-1)$ for an $[[n,k,d]]$ QEC code. For the [[7,1,3]] code, each qubit spends one cycle teleporting, and six waiting for the other teleportations. If p_m is the probability of error for a single qubit during the time to execute a single teleportation, then the probability of no error on one bit during that time is $(1-p_m)^6$ for the [[7,1,3]] code. For an $[[n,k,d]]$ code, the failure probability of that qubit during the serial transfer waiting time is $p'_m = 1 - (1-p_m)^{n-1}$. The probability

of m memory errors is

$$\begin{aligned} p_M(n, m) &= \binom{n}{m} p_m'^m (1 - p_m')^{n-m} \\ &\approx \binom{n}{m} p_m'^m \\ &\approx \binom{n}{m} (n-1) p_m^m. \end{aligned} \quad (9)$$

Combining Equations 9 and 3, we need the two error sources together to generate less than $m = (d+1)/2$ errors. We will constrain the final combined memory and teleportation error rate p_f for the serial link to be similar to the teleportation errors for the parallel link,

$$p_f(n, m) = \sum_{i=0}^m p_M(n, i) p_e(n, m-i) \sim p_e(n, m). \quad (10)$$

For the error codes we are considering, $[[7,1,3]]$ and $[[23,1,7]]$, numeric evaluation for $p_m = p_t/10(n-1)$ gives 25% and 50% increase in failure probability, respectively, compared to the $p_m = 0$ (perfect memory) case. Thus, we can say, very roughly, that a memory failure probability two orders of magnitude less than the failure probability of the teleportation operation will mean that the choice of serial or parallel buses has minimal impact on the overall system error rate.

Although this section has focused on reliability rather than performance, the choice of serial or parallel links also affects performance. It is easy to see that choosing a serial link does not result in a factor of n degradation in system performance when QEC is taken into account. Let t_t be our teleportation time, and t_{LQEC} be the time to perform local error correction. t_t is related to the detector time for measuring the probe beam on the long-distance links, while t_{LQEC} is related to the local qubit measurement time.

If $nt_t \ll t_{LQEC}$, then in accordance with Amdahl's Law the choice also has minimal impact on our overall performance [65]. Moreover, for Shor's algorithm on the quantum multicomputer, we have shown that breaking down the teleportation operation into its component phases of EPR pair creation and the measurement and classical operations allows application-level performance to be relatively independent of the quantum link operation time [8]. Therefore, we recommend using serial links.

VI. SUMMARY

This paper has tackled two important issues in the design of distributed quantum computing systems, both centering around the need to correct errors that occur during teleportation, analyzed in the context of a long but finite computation such as Shor's factoring algorithm. We have shown that a relatively high failure rate for teleportation is tolerable, and that using serial links rather than parallel has only a modest impact on the probability of failure and the performance of the computation.

The results in Table III show that a teleportation error rate (related to the EPR pair infidelity) of $> 1\%$ will allow computations as large as the factoring of a 1,024-bit number

to proceed with a high probability of success. This estimate is for a data encoding of $[[23,1,7]]^i + [[23,1,7]]^o$ on the link. Our analysis supports Steane's recommendation of the $[[23,1,7]]$ code. Replacing one level with the $[[7,1,3]]$ code still allows an error rate of one part in a thousand or better, with a noticeable savings in storage requirements. Of course, we do not have to compute or store data using the same encoded states that we use during data transport, as noted by Thaker *et al.* [60]. In this paper, for simplicity, we have assumed that the system uses only a single choice of encoding.

We have argued that the difference in both performance and reliability between serial and parallel network links will be small for a reasonable set of assumptions. A memory error rate in the time it takes to perform a teleportation at least two orders of magnitude better than the teleportation failure rate results in a 25 – 50% increase in the computation failure rate, an increase we consider acceptable in exchange for the benefits of serial links. Serial links will dramatically simplify our hardware design by reducing the number of required transceiver qubits in each node, and eliminating concerns such as jitter and skew between pairs of conductors or wave guides. Moreover, if we do choose to have multiple transceiver qubits in each node, system performance on some workloads may be boosted more by creating a richer node-to-node interconnect topology than by creating parallel channels between pairs of nodes in a simpler topology.

ACKNOWLEDGMENTS

The authors thank MEXT and QAP for partial support for this research. We thank Kohei M. Itoh for technical help.

REFERENCES

- [1] Lov K. Grover. Quantum teleportation. <http://arXiv.org/quant-ph/9704012>, April 1997.
- [2] R. Cleve and H. Buhrman. Substituting quantum entanglement for communication. *Physical Review A*, 56(2):1201–1204, 1997.
- [3] J.I. Cirac, A. Ekert, S.F. Huelga, and C. Macchiavello. Distributed quantum computation over noisy channels. *Physical Review A*, 59:4249, 1999.
- [4] Ellie D'Hondt. *Distributed quantum computation: A measurement-based approach*. PhD thesis, Vrije Universiteit Brussel, July 2005.
- [5] Daniel K. L. Oi, Simon J. Devitt, and Lloyd C. L. Hollenberg. Scalable error correction in distributed ion trap computers. [quant-ph/0606226](http://arXiv.org/quant-ph/0606226), June 2006.
- [6] A. M. Steane and D. M. Lucas. Quantum computing with trapped ions, atoms, and light. *Fortschritte der Physik*, April 2000. <http://arXiv.org/quant-ph/0004053>.
- [7] Rodney Van Meter and Mark Oskin. Architectural implications of quantum computing technologies. *ACM Journal of Emerging Technologies in Computing Systems*, 2(1):31–63, January 2006.
- [8] Rodney Van Meter, W. J. Munro, Kae Nemoto, and Kohei M. Itoh. Distributed arithmetic on a quantum multicomputer. In *Computer Architecture News, Proc. 33rd Annual International Symposium on Computer Architecture* [66], pages 354–365.
- [9] Chip Elliott, David Pearson, and Gregory Troxel. Quantum cryptography in practice. In *Proc. SIGCOMM 2003*. ACM, ACM, August 2003. <http://arXiv.org/quant-ph/0307049>.
- [10] D. Felinto, C. W. Chou, J. Laurat, H. Schomburg, E. W. de Riedmatten, and H. J. Kimble. Conditional control of the quantum states of remote atomic memories for quantum networking. *Nature Physics*, 2006. doi: 10.1038/nphys450.
- [11] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

- [12] Timothy P. Spiller, William J. Munro, Sean D. Barrett, and Pieter Kok. An introduction to quantum information processing: applications and realisations. *Contemporary Physics*, 46:406, 2005.
- [13] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proc. 35th Symposium on Foundations of Computer Science*, pages 124–134, Los Alamitos, CA, 1994. IEEE Computer Society Press.
- [14] L. Grover. A fast quantum-mechanical algorithm for database search. In *Proc. 28th Annual ACM Symposium on the Theory of Computation*, pages 212–219, 1996. <http://arXiv.org/quant-ph/9605043>.
- [15] D. Deutsch and R. Jozsa. Rapid solution of problems by quantum computation. *Proc. R. Soc. London, Ser. A*, 439:553, 1992.
- [16] Isaac L. Chuang, Lieven M. K. Vandersypen, Xinlan Zhou, Debbie W. Leung, and Seth Lloyd. Experimental realization of a quantum algorithm. *Nature*, 393:143–146, 1998.
- [17] Isaac L. Chuang, Neil Gershenfeld, and Mark Kubinec. Experimental implementation of fast quantum searching. *Physical Review Letters*, 80:2408–2411, 1998.
- [18] Lieven M. K. Vandersypen, Matthias Steffen, Gregory Breyta, Constantino S. Yannoni, Mark H. Sherwood, and Isaac L. Chuang. Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance. *Nature*, 414:883–887, December 2001.
- [19] C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, and W. Wootters. Teleporting an unknown quantum state via dual classical and EPR channels. *Physical Review Letters*, 70:1895–1899, 1993.
- [20] A. Furusawa, J. L. Sørensen, S. L. Braunstein, C. A. Fuchs, H. J. Kimble, and E. S. Polzik. Unconditional quantum teleportation. *Science*, 282(5389):706–709, 1998.
- [21] Richard Jozsa and Noah Linden. On the role of entanglement in quantum computational speedup. *Proc. Royal Soc. London A*, 459:2011–2032, 2003. <http://arXiv.org/quant-ph/0201143>.
- [22] P.J. Love and B.M. Boghosian. Type II quantum algorithms. *Physica A: Statistical Mechanics and its Applications*, 362(1):210–214, 2006.
- [23] Jeffrey Yeepez. Type-II quantum computers. *International Journal of Modern Physics C*, 12(9):1273–1284, 2001.
- [24] Rodney Doyle Van Meter III. *Architecture of a Quantum Multicomputer Optimized for Shor's Factoring Algorithm*. PhD thesis, Keio University, 2006.
- [25] Anocha Yimsiriwattana and Samuel J. Lomonaco Jr. Distributed quantum computing: A distributed Shor algorithm. <http://arxiv.org/quant-ph/0403146>, March 2004.
- [26] Daniel Gottesman and Isaac L. Chuang. Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature*, 402:390–393, 1999.
- [27] W.J. Munro, K. Nemoto, and T.P. Spiller. Weak nonlinearities: a new route to optical quantum computation. *New Journal of Physics*, 7:137, May 2005.
- [28] T. P. Spiller, Kae Nemoto, Samuel L. Braunstein, W. J. Munro, P. van Loock, and G. J. Milburn. Quantum computation by communication. *New Journal of Physics*, 8:30, February 2006.
- [29] K. Nemoto and W.J. Munro. Universal quantum computation on the power of quantum non-demolition measurements. *Physics Letters A*, 344:104, 2005.
- [30] H.-J. Briegel, W. Dür, J.I. Cirac, and P. Zoller. Quantum repeaters: the role of imperfect local operations in quantum communication. *Physical Review Letters*, 81:5932–5935, 1998.
- [31] L. Childress, J.M. Taylor, A.S. Sørensen, and M.D. Lukin. Fault-tolerant quantum repeaters with minimal physical resources and implementations based on single-photon emitters. *Physical Review A*, 72(5):52330, 2005.
- [32] L. Childress, J.M. Taylor, A.S. Sørensen, and M.D. Lukin. Fault-tolerant quantum communication based on solid-state photon emitters. *Physical Review Letters*, 96(7):70504, 2006.
- [33] S. J. van Enk, J. I. Cirac, and P. Zoller. Photonic channels for quantum communication. *Science*, 279(5348):205–208, 1998.
- [34] J.I. Cirac, P. Zoller, H.J. Kimble, and H. Mabuchi. Quantum state transfer and entanglement distribution among distant nodes in a quantum network. *Physical Review Letters*, 78(16):3221–3224, 1997.
- [35] S. Mancini and S. Bose. Engineering an interaction and entanglement between distant atoms. *Physical Review A*, 70(2):22307, 2004.
- [36] L.-M. Duan, B. B. Blinov, D. L. Moehring, and C. Monroe. Scalable trapped ion quantum computation with a probabilistic ion-photon mapping. *Quantum Information and Computation*, 4:165–173, 2004.
- [37] L.M. Duan, B. Wang, and HJ Kimble. Robust quantum gates on neutral atoms with cavity-assisted photon scattering. *Physical Review A*, 72(3):32333, 2005.
- [38] S. J. van Enk, J. I. Cirac, and P. Zoller. Ideal quantum communication over noisy channels: A quantum optical implementation. *Phys. Rev. Lett.*, 78(22):4293–4296, Jun 1997.
- [39] E. Waks and J. Vuckovic. Dipole induced transparency in drop-filter cavity-waveguide systems. *Physical Review Letters*, 96(15):153601, 2006.
- [40] Wang Yao, Ren-Bao Liu, and L. J. Sham. Theory of control of the spin-photon interface for quantum networks. *Physical Review Letters*, 95:030504, 2005.
- [41] W. Dür, H.-J. Briegel, J. I. Cirac, and P. Zoller. Quantum repeaters based on entanglement purification. *Physical Review A*, 59(1):169–181, Jan 1999.
- [42] Zeng-Bing Chen, Bo Zhao, Joerg Schmiedmayer, and Jian-Wei Pan. Fault-tolerant quantum repeater with atomic ensembles and linear optics. [quant-ph/0609151](http://arXiv.org/quant-ph/0609151), September 2006.
- [43] L.M. Duan, M.D. Lukin, J.I. Cirac, and P. Zoller. Long-distance quantum communication with atomic ensembles and linear optics. *Nature*, 414:413–418, 2001.
- [44] Bo Zhao, Zeng-Bing Chen, Yu-Ao Chen, Joerg Schmiedmayer, and Jian-Wei Pan. Robust long-distance quantum communication with atomic ensembles and linear optics. [quant-ph/0609154](http://arXiv.org/quant-ph/0609154), September 2006.
- [45] L. Jiang, J. M. Taylor, and M. D. Lukin. A fast and robust approach to long-distance quantum communication with atomic ensembles. [quant-ph/0609236](http://arXiv.org/quant-ph/0609236), September 2006.
- [46] Rodney Van Meter and Kohei M. Itoh. Fast quantum modular exponentiation. *Physical Review A*, 71(5):052320, May 2005.
- [47] David Beckman, Amalavoyal N. Chari, Srikrishna Devabhaktuni, and John Preskill. Efficient networks for quantum factoring. *Phys. Rev. A*, 54:1034–1063, 1996. <http://arXiv.org/quant-ph/9602016>.
- [48] Vlatko Vedral, Adriano Barenco, and Artur Ekert. Quantum networks for elementary arithmetic operations. *Phys. Rev. A*, 54:147–153, 1996. <http://arXiv.org/quant-ph/9511018>.
- [49] Thomas G. Draper, Samuel A. Kutin, Eric M. Rains, and Krysta M. Svore. A logarithmic-depth quantum carry-lookahead adder. <http://arXiv.org/quant-ph/0406142>, June 2004.
- [50] Steve A. Cuccaro, Thomas G. Draper, Samuel A. Kutin, and David Petrie Moulton. A new quantum ripple-carry addition circuit. <http://arXiv.org/quant-ph/0410184>, October 2004.
- [51] Peter W. Shor. Fault-tolerant quantum computation. In *Proc. 37th Symposium on Foundations of Computer Science*, pages 56–65, Los Alamitos, CA, 1996. IEEE Computer Society Press.
- [52] Andrew Steane. Error correcting codes in quantum theory. *Physical Review Letters*, 77:793–797, 1996.
- [53] A. R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54:1098–1105, 1996.
- [54] Charles H. Bennett, David P. DiVincenzo, John A. Smolin, and William K. Wootters. Mixed-state entanglement and quantum error correction. *Physical Review A*, 54:3824–3851, 1996.
- [55] Raymond Laflamme, Cesar Miquel, Juan Pablo Paz, and Wojciech Hubert Zurek. Perfect quantum error correcting code. *Physical Review Letters*, 77:198–201, 1996.
- [56] Andrew M. Steane. Overhead and noise threshold of fault-tolerant quantum error correction. *Physical Review A*, 68:042322, 2003.
- [57] Todd Brun, Igor Devetak, and Min-Hsiu Hsieh. Correcting Quantum Errors with Entanglement. *Science*, 314(5798):436–439, 2006.
- [58] Fumiko Yamaguchi, Kae Nemoto, and William J. Munro. Quantum error correction via robust probe modes. *Physical Review A*, 73:060302, June 2006.
- [59] D. Bacon. Operator quantum error-correcting subsystems for self-correcting quantum memories. *Physical Review A*, 73(1):12340, 2006.
- [60] Darshan D. Thaker, Tzvetan Metodiev, Andrew Cross, Isaac Chuang, and Frederic T. Chong. CQLA: Matching density to exploitable parallelism in quantum computing. In *Computer Architecture News, Proc. 33rd Annual International Symposium on Computer Architecture* [66].
- [61] Charles H. Bennett, Herbert J. Bernstein, Sandu Popescu, and Benjamin Schumacher. Concentrating partial entanglement by local operations. *Physical Review A*, 53:2046, 1996.
- [62] L. Hartmann, B. Kraus, H.-J. Briegel, and W. Dür. On the role of memory errors in quantum repeaters. [quant-ph/0610113](http://arXiv.org/quant-ph/0610113), October 2006.
- [63] Dorit Aharonov and Michael Ben-Or. Fault-tolerant quantum computation with constant error rate. <http://arXiv.org/quant-ph/9906129>, June 1999. extended version of STOC 1997 paper.
- [64] John Preskill. Reliable quantum computers. *Proc. Roy. Soc. Lond. A*, 454:385–410, 1998.

- [65] Gene Amdahl. Validity of the single processor approach to achieving large-scale computing capabilities. In *AFIPS Conference Proceedings*, pages 483–485, 1967.
- [66] ACM. *Computer Architecture News, Proc. 33rd Annual International Symposium on Computer Architecture*, June 2006.